REMARKS/ARGUMENTS

Claims 1, 3-8, 10, 12-17 and 19-23 are pending in the present application. Claims 1, 3-6, 10, 12-14, 17 and 19-21 have been amended, and Claims 2, 9, 11 and 18 have been cancelled, herewith. Reconsideration of the claims is respectfully requested.

I. Claim Objection

Claim 17 was objected to, with the Examiner suggesting the elimination of 'the steps of' in the preamble, and Applicants have amended such claim accordingly.

Therefore, the objection to the claims has been overcome.

II. 35 U.S.C. § 101

Claims 1-5 and 10-13 stand rejected under 35 U.S.C. § 101 as being directed towards non-statutory subject matter. This rejection is respectfully traversed.

Applicants have amended Claims 1 and 10 to include features previously recited in Claim 9 (which is thus being cancelled herewith, without prejudice or disclaimer).

Therefore, the rejection of Claims 1-5 and 10-13 under 35 U.S.C. § 101 has been overcome.

III. 35 U.S.C. § 102, Anticipation

Claims 1-3, 5-11, 13-19 and 21-23 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Coha et al., US Publication No. 2003/0182597 – now issued as US Patent No. 6,804,691 (hereinafter Coha). This rejection is respectfully traversed.

Generally speaking, the present claims are directed to techniques for determining the likely effects on program performance resulting from modification of a program, prior to the program being modified to achieve performance improvements, if the effects on program performance are determined to be favorable to performance improvements. Per the teachings of the cited reference, various parameters fed into a garbage collector simulator are output to a user in graphical form, and these parameters can be manually modified, the simulator re-run, and the parameters are again presented to the user in graphical form where they can again be manually modified and the simulator re-run. Once satisfactory results are obtained through this manual, parameter-adjusting trial-and-error simulation process, the heap parameters are inserted into the Java virtual machine (JVM). In effect, per Claim 1 an automated algorithmic approach to parameter tuning for an application program is provided, whereas per the teachings of the cited reference a manually repeated 'trial-and-error' approach to optimizing garbage collection is provided. This manual approach to tuning the operating environment used by an application program, as taught by Coha, is improved by the automated algorithmic approach provided by the present invention.

Specifically with respect to Claim 1, Applicants have amended such claim to include features previously recited in Claim 9 to address the 35 U.S.C. § 101 claim rejection. In rejecting Claim 9 (whose features are now included in amended Claim 1), the Examiner states that Coha teaches the modifying of the Java computer program at Figure 2A. Applicants urge that, instead, Coha teaches the modification of a Java Virtual Machine (JVM) (see, e.g. Coha Figure 2A, block 260), which is not a Java computer program but instead is a *runtime environment* that is used with a Java computer program to allow the Java computer program to be written in a hardware independent fashion (Specification page 11, line 20 – page 12, line 1; Coha page 1, paragraphs [0002] and [0003]). By analogy with respect to a DVD player, the claimed computer program is akin to the information stored on a DVD media that is to be played by the DVD player, whereas the Coha-taught JVM is akin to the operating environment provided by the circuitry provided by the embedded electronics within the DVD player (i.e. the environment that is used to play the DVD). By further analogy, the claimed computer program is akin to an application program executed on a computer such as a word processing program or email program, whereas the Coha-taught JVM is akin to the operating system or BIOS that provides the operating environment for which the computer program can be executed. Claim 1 is directed to techniques for improving performance on the actual JVM computer program itself that is processed by the JVM, whereas Coha addresses changing the JVM operating environment. Thus, as every element recited in Claim 1 is not identically shown in a single reference, it is urged that Claim 1 is not anticipated by the cited reference.

Further with respect to Claim 1, the cited reference does not provide any type of *garbage collection cost <u>estimation</u>* to the performance of a program, but instead runs a garbage collection simulation where results of such simulation are graphically presented to a user for possible manual override or fine-tuning of *garbage collections parameters and policies* (Coha page 1, paragraphs [0005] and [0011], page 2, paragraph [0022], and particularly page 3, paragraph [0040]). Such fine-tuning of garbage collection parameters used by the JVM itself does not provide any type of garbage collection cost estimation with respect to the actual Java application program. This can also be seen by Coha's Figure 3, which is the 'results' screen presented to a user as a result of a garbage collection simulation run. This graphical depiction provides no information pertaining to a *<u>cost estimation</u> to the performance of a Java computer application program*. Thus, as every element recited in Claim 1 is not identically shown in a single reference, it is urged that Claim 1 is not anticipated by the cited reference.

Still further with respect to Claim 1, Applicants have amended such claim to include features previously recited in Claim 2 (which is thus being cancelled herewith, without prejudice or disclaimer).

As amended, Claim 1 recites "wherein a cost of garbage collection to program performance of the Java computer application program is estimated using a *duration* of an average garbage collection event and a *frequency* of garbage collection events" (emphasis added by Applicants). As can be seen, both (1) a *duration* of an average garbage collection event and (2) a *frequency* of garbage collection events are used when estimating the cost of garbage collection to program performance. The cited reference does not teach use of both duration of an average garbage collection event and *frequency of garbage collection events*. In rejecting this aspect of Claim 2 (whose features are now a part of amended Claim 1), the Examiner cites Coha's teaching at paragraphs [0037], [0058]-[0059] and [0063] as teaching use of both a duration and frequency of garbage collection events. At paragraph [0037], Coha states:

> "[0037] Data plot 305 comprises an x-axis 310 comprising the overall application run time in seconds and a y-axis 315 comprising the run time of a specific garbage collection routine in seconds. Furthermore, data plot 305 illustrates the duration of four types of garbage collection routines: scavenge, system.gc, old full, and old too full. Specifically, data plot 305 illustrates when a garbage collection occurred, what type of garbage collection occurred, and how long it took to run. Overall timing information, such as the average scavenge garbage collection duration and the average full garbage collection duration are shown in timing information 320."

At paragraph [0058], Coha states:

> "[0058] the intervals between garbage collection;"

At paragraph [0059], Coha states:

> "[0059] the duration of each of the individual collections;"

At paragraph [0063], Coha states:

> "[0063] the duration of each individual garbage collection;"

As can be seen, while Coha may mention the display of parameters associated with the duration of garbage collection events, there is no mention of any use of *frequency of garbage collection events*. Perhaps the Examiner is improperly interpreting 'the intervals between' garbage collection events (Coha paragraph [0058]) as being equivalent to the claimed frequency of events. Applicants urge that the intervals *between* garbage collection provide no information as to the actual frequency that the garbage collection events themselves occur. By analogy, a car collector such as Jay Leno may have numerous automobiles stored in a garage that are infrequently driven. Such a car collector may need to track the period of time (i.e. interval) since it was last driven (i.e. the event) in order to determine if the battery should be charged before driving the car. For example, if the car has not been driven in three (3) months the car collector may charge the battery prior to driving it to ensure the battery functions properly. Such three (3) month period of time since it was last driven (the 'interval' between drives) provides no information as to the frequency at which the car has been driven (the 'driving' event being analogous to the claimed garbage collection event). Following this car analogy, assume the 'event' is an oil change, and a car owner tracks the period of time between oil changes (the 'interval' between the oil change 'event'). The *period of time* *between* oil changes (by analogy, the period of time between garbage collection events) provides no information with respect to the frequency at which the oil change events occur. Quite simply, 'intervals' *between* events (as taught by Coha) provides no information with respect to the 'frequency' at which *the events themselves occur* (as claimed). Thus, while Coha may teach a determination with respect to the duration of a garbage collection event, Coha does not teach (1) any determination with respect to *both duration and frequency* of garbage collection events, or (2) the *use* of both duration and frequency of garbage collection events *to estimate garbage collection cost to the performance of a program*, as recited in amended Claim 1. Thus, it is urged that Claim 1 is not anticipated by the cited reference as every element recited therein is not identically shown in a single reference[1].

Applicants initially traverse the rejection of Claims 3 and 5-9 for reasons given above with respect to Claim 1 (of which Claims 3 and 5-9 depend upon).

Further with respect to Claim 5, it is urged that the cited reference does not teach the claimed feature of "wherein the frequency depends on the rate of object creation, the heap fragmentation, the size of the heap, and the garbage collection policy". As can be seen, the *frequency* that is used in the cost estimation depends upon four explicitly enumerated items: (1) the rate of object creation, (2) the heap fragmentation, (3) the size of the heap, and (4) the garbage collection policy. In rejecting the 'rate of object creation' aspect of Claim 5, the Examiner cites Coha paragraph [0038], [0056], [0047] and [0054] as teaching this claimed feature. Applicants urge error, as none of these cited passages teach (1) a rate of object creation or (2) a frequency that is used in the cost estimation that depends upon the use of a rate of

[1] For a prior art reference to anticipate in terms of 35 U.S.C. 102, every element of the claimed invention must be identically shown in a single reference. *In re Bond*, 910 F.2d 831, 15 USPQ2d 1566 (Fed. Cir. 1990) (emphasis added by Applicants).

object creation. While Coha alludes to a creation rate at paragraph [0038], there is no mention of what this 'creation' pertains to. However, Coha does in fact describe what this creation rate pertains to at paragraphs [0055], [0056] and Claim 22, and it is the rate of memory allocation and the rate at which memory becomes unreferenced. Neither a memory creation rate nor a unreferenced memory rate teaches or suggests a *rate of object creation*, as expressly recited in Claim 5, nor a frequency that is used in the cost estimation that depends upon the use of such (missing) rate of object creation. Thus, it is further urged that Claim 5 is not anticipated by the cited reference as there are additional claimed features that are not taught by the cited reference.

Further with respect to Claim 6, and for similar reasons to those described above with respect to Claim 1, the cited Coha reference does not teach "wherein the Java computer application program is changed by reducing memory from a footprint of the Java computer application program". The cited reference does not teach any technique that reduces memory from a footprint of a Java program. Instead, garbage collection policies used by the JVM environment are fine-tuned (Coha paragraph [0004] et seq). Thus, it is further urged that Claim 6 is not anticipated by the cited reference as there are additional claimed features that are not taught by the cited reference.

Applicants traverse the rejection of Claims 10-11, 13-19 and 21-23 for similar reasons to those given above with respect to Claim 1.

Applicants further traverse the rejection of Claims 13 and 21 for similar reasons to the further reasons given above with respect to Claim 5.

Applicants further traverse the rejection of Claim 14 for similar reasons to the further reasons given above with respect to Claim 6.

Therefore, the rejection of Claims 1-3, 5-11, 13-19 and 21-23 under 35 U.S.C. § 102(e) has been overcome.

IV. 35 U.S.C. § 103, Obviousness

Claims 4, 12 and 20 stand rejected under 35 U.S.C. § 103 as being unpatentable over Coha et al., US Publication No. 2003/0182597, in view of Sumit Chawla, 'Fine-tuning Java Garbage collection performance', 01 Jan 2003, pp. 1-10; (hereinafter Chawla – url: http://www.128.ibm.com/developerworks/ibm/library/i-gctroub/). This rejection is respectfully traversed.

Applicants initially traverse the rejection of Claim 4 (and similarly for Claims 12 and 20) for similar reasons to those given above with respect to Claim 1 (of which Claim 4 depends upon), and urge that the newly cited Chawla reference does not overcome the teaching deficiencies identified above with respect to Claim 1.

Further with respect to Claim 4, it is urged that the newly cited Chawla reference does not teach or otherwise suggest that a *duration* that is used in estimating a cost of garbage collection to program performance depends upon a number of finalizers that must be executed. Chawla in fact expressly teaches away from any similar type of/related claimed feature, as Chawla explicitly disdains any use of finalizers. For example, as stated by Chawla at page 6:

> "The culprit here is the number of objects that had to be finalized. Finalizers are a bad idea anyway, and though they are unavoidable in certain circumstances, *they should be used <u>only</u> as a last resort and <u>only</u> for actions that cannot be done elsewhere*" (emphasis added by Applicants).

As can be seen, the only use of finalizers described by Chawla is that they should be used *only as a last resort* and *only for actions that cannot be done elsewhere*. The claimed use of finalizers as expressly recited in Claim 4 does not meet either of these 'only' conditions as taught by Chawla, and thus Chawla does not teach or otherwise suggest the claimed feature of that a *<u>duration</u> that is <u>used</u> in estimating a cost of garbage collection* to program performance *<u>depends upon</u> a number of finalizers* that must be executed. Thus, it is urged that the Examiner has failed to properly establish a prima facie showing of obviousness with respect to Claim 4[2]. Accordingly, the burden has not shifted to Applicants to rebut such improper obviousness assertion[3]. In addition, as a proper prima facie showing of obviousness has not been established by the Examiner, Claim 4 (and similarly for Claims 12 and 20) has been erroneously rejected under 35 U.S.C. § 103[4].

Therefore, the rejection of Claims 4, 12 and 20 under 35 U.S.C. § 103 has been overcome.

[2] To establish prima facie obviousness of a claimed invention, all of the claim limitations must be taught or suggested by the prior art. MPEP 2143.03. *See also, In re Royka*, 490 F.2d 580 (C.C.P.A. 1974).

[3] In rejecting claims under 35 U.S.C. Section 103, the examiner bears the initial burden of presenting a prima facie case of obviousness. *In re Oetiker*, 977 F.2d 1443, 1445, 24 USPQ2d 1443, 1444 (Fed. Cir. 1992). Only if that burden is met, does the burden of coming forward with evidence or argument shift to the applicant. *Id.*

[4] If the examiner fails to establish a prima facie case, the rejection is improper and will be overturned. *In re Fine*, 837 F.2d 1071, 1074, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988).

V. Conclusion

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: May 7, 2007

Respectfully submitted,

/Wayne P. Bailey/

Wayne P. Bailey
Reg. No. 34,289
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicant